Research, MIT Press, ...
in Kyoto, Japan, Aug, 1984

ON THE DEVELOPMENT OF HIGH PERFORMANCE ADAPTIVE CONTROL ALGORITHMS FOR ROBOTIC MANIPULATORS

Steven Dubowsky, Professor
Roy Kornbluh, Graduate Research Assistant
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139
U.S.A.

NAG1-389
IN-37-CR
8P

Complex industrial applications of robotic manipulators are beyond the capabilities of the relatively simple constant-gain linear control systems commonly used for current industrial manipulators. They compensate poorly for the nonlinear dynamic characteristics of manipulators and environmental changes. This paper presents the development of an adaptive control algorithm which adjusts, in real time, the control system parameters to compensate for manipulator nonlinear characteristics and changes in its environment and payload. Experimental results are also presented for two devices which show that the algorithm is practical and capable of producing significantly improved systems dynamic performance.

Introduction

Current commercial manipulators use relatively simple constant gain linear feedback control systems. While such control systems are reliable and easy to implement and design, their performance is rather limited. They compensate poorly for the nonlinear manipulator character and are sensitive to environmental changes, mechanical flexibility and actuator limitations. Hence, many complex industrial applications are beyond the capabilities of current industrial robotic manipulators [National Reseach Council].

A key to improved manipulator performance is the development of better control algorithms. In recent years, researchers have proposed a number of control methods that may substantially improve system performance. However, their value is almost always demonstrated analytically or by simulation, and only for idealized systems that may fail to consider important system factors which limit and degrade control system performance. These factors include manipulator friction, sensor noise, unknown system parameters and disturbances. In this paper an adaptive control algorithm for the motion control of robotic manipulators is presented which has been shown by simulation to signficantly improve system performance. Experimental results are presented which demonstrate the value of the algorithm on a laboratory robotic device and a commercial manipulator system.

The Literature

Many of the approaches suggested for improved motion control are based on explicit representations of the manipulator's nonlinear dynamics used in a feedforward manner [Raibert and Horn,1978].Linear feedback is provided to compensate for model errors and disturbances. Nonlinear feedback control laws have also been proposed [Freund, 1982; Luh, Walker and Paul 1980]. Optimal control theory has been applied linearized manipulator models [Kahn and Roth,1971], and more recently to manipulators with full nonlinear dynamics [Bobrow, Dubowsky and Gibson, 1983]. An important aspect of the above control approaches is that they require accurate dynamic models of the manipulator which are often difficult to obtain. A minipulator's dynamic behavior will vary over time due to such factors as varaiable joint friction and payload mass.

Adaptive control techniques have been proposed which attempt to adjust control system characteristics to compensate for changing dynamic properties based on measured performance rather than a detailed knowledge of the manipulator dynamics. Within adaptive control there are two fundamental approaches. The first is Learning Model Adaptive Control (LMAC), in which an improved model of the manipulator is obtained by on-line parameter identification techniques [Koivo and Guo,1983], and is then used in the feedback control of manipulators. The second approach is called Model Referenced Adaptive Control (MRAC). The controller is adjusted so that the closed loop behavior of a system matches that of a preselected model according to some criterion [Landau, 1974]. It is possible to have adaptation schemes which employ aspects of both approaches [LeBorgne, Ibarra, and Espiau 1981].

The manipulator model assumed in an adaptive control approach may range from a relatively simple linear uncoupled differential equation [Dubowsky and DesForges, 1979; Koivo, 1983] to more complex models of the manipulator dynamics [Vukobratovic and Kiracanski, 1982]. For high speed manipulator motion or control in task space there may be significant dynamic coupling between the joints which suggest mult-input-multi-output (MIMO) approaches [Takegaki and Arimoto, 1981]. However, studies have found that uncoupled single-input-single-output (SISO) algorithms can handle dynamic interaction without significant performance degradation [Koivo and Guo, 1983; LeBorgne, Ibarra and Espiau, 1981]. Adaptive

(NASA-CR-186389) ON THE DEVELOPMENT OF HIGH PERFORMANCE ADAPTIVE CONTROL ALGORITHMS FOR ROBOTIC MANIPULATORS (MIT) 8 p
N90-70716
Unclas
00/37 0270014

algorithms have been proposed for manipulator control which employ non-adaptive nonlinear control in addition to adaptation [Lee, Ching and Lee, 1984].

MRAC algorithms can be developed in several ways. One common approach relies on a global stability theorem, such as Popov's Hyperstability Theorem. [Horowitz and Tomizuka, 1980]. Some MRAC algorithms, based on this approach, are unstable in practical applications due to unmodelled higher order dynamics and sensor noise [Rhors, 1983]. In this work MIMO algorithms are developed by applying the method of Steepest Descent to minimizing a cost function of the error between the model and the system.

Analytical Development

The manipulator dynamics, including its control, can be described by the nonlinear differential equation;

$$\dot{\underline{x}} = \underline{f}_s(\underline{x}, \underline{\alpha}, \underline{r}, \underline{d}) \qquad (1)$$

where $\underline{x}$ is the system state vector$(x_1,\ldots x_n)$,
$\underline{\alpha}$ is a system parameter vector$(\alpha_1,\ldots \alpha_m)$,
$\underline{r}$ is the input vector$(r_1,\ldots r_n)$, and
$\underline{d}$ is a disturbance vector$(d_1,\ldots d_n)$.

The $\underline{\alpha}$ vector is a function of x, and the control matrix, $K(t)$, which is adjusted by the adaptive algorithm. The model state equation for the system to track is written as:

$$\dot{\underline{y}} = \underline{f}_m(\underline{y}, \underline{a}, \underline{r}) \qquad (2)$$

where $\underline{y}$ is the model state vector$(y_1,\ldots y_n)$ and
$\underline{a}$ is a model parameter vector$(a_1,\ldots a_m)$.

The error between the response of the model and the system is defined by the vector:

$$\underline{e}(t) = \underline{y}(t) - \underline{x}(t) \qquad (3)$$

The objective of the algorithm is to manipulate $K(t)$, and hence $\underline{\alpha}$, in such a way as to drive $\underline{e}(t)$ to zero, and thereby match the system response to that of the model (i.e. $\underline{\alpha} = \underline{a}$), neglecting unmodelled disturbances. The development uses the method of Sensitivity-Steepest Descent [Donalson and Leondes, 1963; Brogan, 1974]. This method uses the parameters of the model rather than the system. Hence, it is computationally less burdensome than methods which evaluate complex, nonlinear models. It has also been shown to have very good noise rejection properties [Papadopoulos, 1983].

In this method, a scalar cost function is formulated as:

$$V(\underline{e}) = 1/2\ \underline{e}^T Q \underline{e} \qquad (4)$$

where Q is called the adaptive gain matirx. The change of $\underline{\alpha}$ in time, such that $V(\underline{e})$ moves along the steepest descent path to a minimum is given by:

$$\dot{\underline{\alpha}} = -k\ \frac{\partial V}{\partial \underline{\alpha}} = -k\ \frac{\partial \underline{e}}{\partial \underline{\alpha}}^T Q\ \underline{e} \qquad (5)$$

where $\partial \underline{e}/\partial \underline{\alpha}$ is an n by m matrix whose i,j element is $\partial e_i/\partial \alpha_j$; Q is an n by n symmetric matrix, and k is a positive scalar quantity. However, the partial derivative $\partial \underline{e}/\partial \underline{\alpha}$ in equation (5) cannot in general be calculated [Donalson, 1963; Dubowsky and DesForges, 1979]. It is approximated as follows:

$$\frac{\partial \underline{e}}{\partial \underline{\alpha}} = \frac{\partial(\underline{y}-\underline{x})}{\partial \underline{\alpha}} = \frac{-\partial \underline{x}}{\partial \underline{\alpha}} = \frac{\partial \underline{y}}{\partial \underline{a}} \equiv \Lambda \qquad (6)$$

The n by m matrix Λ, called the sensitivity matarix, can be obtained from the partial derivative of equation (2) with respect to a as follows:

$$\frac{\partial \dot{\underline{y}}}{\partial \underline{a}} = \frac{\partial \underline{f}_m}{\partial \underline{y}}\ \frac{\partial \underline{y}}{\partial \underline{a}} + \frac{\partial \underline{f}_m}{\partial \underline{a}} \qquad (7)$$

Assuming that $\partial \dot{\underline{y}}/\partial \underline{a} = d[\partial y/\partial \underline{a}]/dt$ and substituting the definition of Λ from equation (6) into equation (7) yields the matrix sensitivity equation:

$$\dot{\Lambda} = \frac{\partial \underline{f}_m}{\partial \underline{y}}\ \Lambda + \frac{\partial \underline{f}_m}{\partial \underline{a}} \qquad (8)$$

The adaption law, equation (5), can be written:

$$\dot{\underline{\alpha}} = -k\ \Lambda^T Q \underline{e}(t) \qquad (9)$$

$V(\underline{e})$ has the appearance of a Lyapunov function and [Papdopoulos, 1983; Brogan 1974] have suggested that this fact can be used to show that the method leads to a stable design. The stability of the method has been investigated for the continuous time case [Dubowsky and DesForges, 1979] and the discrete time case, including the effects of computational delays [Dubowsky, 1981].

Given $\dot{\underline{\alpha}}$, from equation (9), and assuming that the unknown parameters of the system change slowly, it is possilbe to solve for the time rate of change of the control gains required to satisfy equation (9), as seen in the following applicaion. A block diagram, Fig. 1, illustrates the above algorithm.

Consider the two degree-of-freedom robotic positioning device shown in Fig. 2. The closed loop system state equation,(1), for this device can be written as:

$$\dot{\underline{x}} = A(\underline{x})\underline{x} + B(\underline{x})\underline{r} + c(\underline{x})\underline{d} \qquad (10)$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -k_{x11}\alpha_1 + k_{x21}\alpha_2 & -k_{x12}\alpha_1 - k_{x22}\alpha_2 + \alpha_3 & -k_{x13}\alpha_1 + k_{x23}\alpha_2 & -k_{x14}\alpha_1 + k_{x24}\alpha_2 + \alpha_4 \\ 0 & 0 & 0 & 1 \\ k_{x11}\alpha_5 - k_{x21}\alpha_6 & k_{x12}\alpha_5 - k_{x22}\alpha_5 + \alpha_7 & k_{x13}\alpha_5 - k_{x23}\alpha_6 & k_{x14}\alpha_5 - k_{x24}\alpha_6 + \alpha_8 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ k_{r11}\alpha_1 - k_{r21}\alpha_2 & k_{r12}\alpha_1 - k_{r22}\alpha_2 \\ -k_{r11}\alpha_5 + k_{r21}\alpha_6 & -k_{r12}\alpha_5 + k_{r22}\alpha_5 \end{bmatrix} \qquad C = \begin{bmatrix} 0 & 0 \\ \alpha_1 & -\alpha_2 \\ 0 & 0 \\ 0 & 0 \\ -\alpha_5 & \alpha_6 \end{bmatrix}$$

and x_1, x_2, x_3 and x_4 are equal to $\theta_1, \dot{\theta}_1, \theta_2$ and $\dot{\theta}_2$ respectively. The $\alpha_1(\underline{x})$ to $\alpha_8(\underline{x})$ terms are nonlinear functions of the state of the system and the payload it carries. Among other factors, they model the effects of the inter-axis dynamic coupling joint friction and changing inertia due to payload and system geometry. K_r is a two by two matrix with elements k_{rij} and K_x is a two by four control gain matrix with elements k_{xij}. To provide unity position feedback k_{r11} was set equal to k_{x11} and k_{r22} equal to k_{x23}. The unknown parameter vector for the system becomes:

$$\underline{a}(\underline{x},K_x,K_r)=[-k_{x11}\dot{a}_1+k_{x21}\dot{a}_2,\ -k_{x12}\dot{a}_2+k_{x22}\dot{a}_2$$

$$-k_{x13}\dot{a}_1+k_{x23}\dot{a}_2,\ -k_{x14}\dot{a}_1+k_{x24}\dot{a}_2+\dot{a}_3,\ k_{x11}\dot{a}_5-k_{x21}\dot{a}_6, \qquad (11)$$

$$k_{x12}\dot{a}_5-k_{x22}\dot{a}_6+\dot{a}_7,\ k_{x13}\dot{a}_5-k_{x23}\dot{a}_5,\ k_{x14}\dot{a}_5-k_{x24}\dot{a}_6-\dot{a}_3]$$

The model selected is that of a linear constant coefficient dynamic system, or:

$$\begin{bmatrix}\dot{y}_1\\ \dot{y}_2\\ \dot{y}_3\\ \dot{y}_4\end{bmatrix}=\begin{bmatrix}0&1&0&0\\ a_1&a_2&a_3&a_4\\ 0&0&0&1\\ a_5&a_6&a_7&a_8\end{bmatrix}\begin{bmatrix}y_1\\ y_2\\ y_3\\ y_4\end{bmatrix}+\begin{bmatrix}0&0\\ -a_1&0\\ 0&0\\ 0&-a_7\end{bmatrix}\begin{bmatrix}r_1\\ r_2\end{bmatrix} \qquad (12).$$

Choosing a_3, a_4, a_5 and a_6 to be zero, the model becomes uncoupled. The model has zero elements in locations where the system does not, due to its assumed decoupled structure. One of the tasks of the adaptive algorithm is to change the k_x's and k_r's as a function of time so as to make these terms approach zero, while making the other terms approximate the selected model constants.

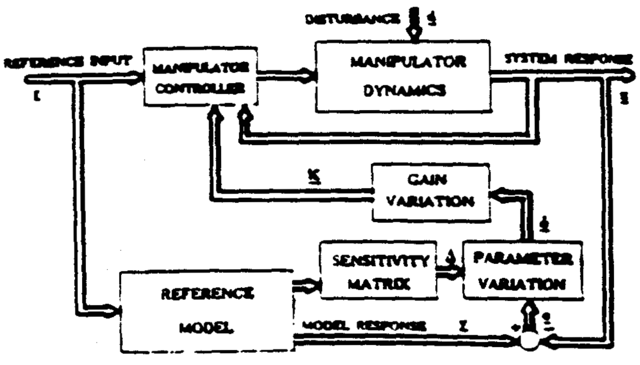

Fig. 1. MIMO Adaptive Algorithm.

It is not always possible to do this exactly, but only in a manner in which $V(\underline{e})$ will be minimized. The vector of selected model parameters is

$$\underline{a}=[\underline{a}_1,\underline{a}_2,\underline{a}_3,\underline{a}_4,\underline{a}_5,\underline{a}_6,\underline{a}_7,\underline{a}_8]^T \qquad (13)$$

Using equation (12) to evaluate equation (8) results in the following matrix sensitivity equation:

$$\dot{\Lambda}=\begin{bmatrix}0&1&0&0\\ a_1&a_2&0&0\\ 0&0&0&1\\ 0&0&a_6&a_7\end{bmatrix}\Lambda+\begin{bmatrix}0&0&0&0&0&0&0&0\\ y_1-r_1&y_2&y_3-r_2&y_4&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&y_1-r_1&y_2&y_3-y_2&y_4\end{bmatrix} \qquad (14)$$

The response of the model from the solution of equation (12), to any input becomes the forcing terms in equation (14). The solution of equation (14) yields $\Lambda(t)$ which can be used in equation (9) with the measured errors to find $\dot{\underline{a}}(t)$. The final task in the application of the algorithm is to calculate the $K_x(t)$ which will produce the desired $\underline{a}(t)$.

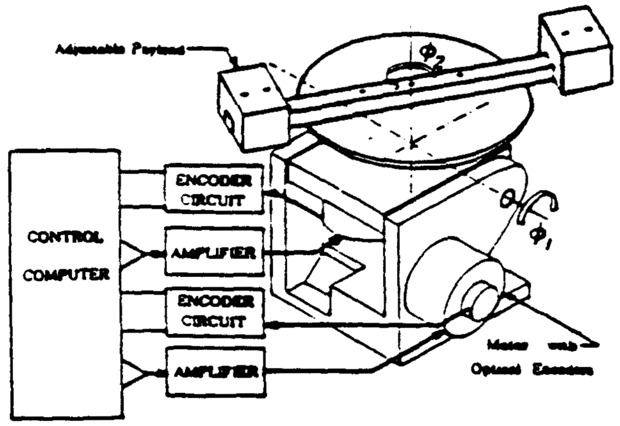

Fig. 2. Two Degree of Freedom Robotic Positioning Device.

Following [Donalson and Leondes, 1963], it is assumed that the parameters of the system, the $\underline{\alpha}$'s, change slowly as compared to the adaption process. It has been shown that while this assumption is useful in developing this form of adaptive algorithm, in practice, it is not necessary for adequate performance of the algorithm [Dubowsky and Desforges, 1979]. With this assumption and the assumption that the adaptive algorithm causes the system parameters to remain close to the model, differentiating (11) yields:

$$\dot{k}_{x11}=\frac{k_{x13}\dot{a}_5}{a_7}+\frac{k_{x11}\dot{a}_1}{a_1} \qquad \dot{k}_{x21}=\frac{k_{x23}\dot{a}_5}{a_7}+\frac{k_{x21}\dot{a}_1}{a_1}$$

$$\dot{k}_{x12}=\frac{k_{x13}\dot{a}_6}{a_7}+\frac{k_{x11}\dot{a}_2}{a_1} \qquad \dot{k}_{x22}=\frac{k_{x23}\dot{a}_6}{a_7}+\frac{k_{x21}\dot{a}_2}{a_1}$$

$$\dot{k}_{x13}=\frac{k_{x13}\dot{a}_7}{a_7}+\frac{k_{x11}\dot{a}_3}{a_1} \qquad \dot{k}_{x23}=\frac{k_{x23}\dot{a}_7}{a_7}+\frac{k_{x21}\dot{a}_3}{a_1}$$

$$\dot{k}_{x14}=\frac{k_{x13}\dot{a}_8}{a_7}+\frac{k_{x11}\dot{a}_4}{a_1} \qquad \dot{k}_{x24}=\frac{k_{x23}\dot{a}_8}{a_7}+\frac{k_{x21}\dot{a}_4}{a_1}$$

These equations can be integrated forward in time to obtain the time varying feedback gains for the system which allows it to follow the behavior of the selected model.

The algorithm equations were put in discrete form for digital computer implementation as described in [Dubowsky, 1981]. Additional simplificiations to the algorithm were made for use in the experimental studies. The dynamic coupling between joints was neglected and simple viscous damping was assumed as each joint. The controller at each joint provides Proportional plus Derivative (PD) control. Following the same procedure a Proportional Integral Derivative (PID) MRAC was also developed. The PID structure used was the "PDF" type [Phelan, 1977]. In the experimental algorithms Q was chosen so that there were no coupled terms in V(e) and hence only two adaptive gains needed to be considered for each joint. These gains were qpi and qvi. The elements of Q can be written as combinations of these two.

Experiments and Simulations

The algorithms were tested on two different devices. One was a two degree-of freedom rotary positioning device (see Fig. 2). The other was a PUMA 500 robotic manipulator (manufactured by Unimation Inc.; see Fig. 3). The two systems are quite similar. Both have geared revolute joints driven by DC servo-motors which are powered by current source analog amplifiers. The dynamic properties of both systems were determined from dynamic tests. Also, both use optical encoders mounted on the motor shafts to provide position and velocity information. The encoder outputs are passed to a circuit which converts the encoder pulses to a sixteen-bit positon count for the control computer. The amplifiers are commanded using an eight-bit digital signal through a D/A converter. A PDP-11/23 minicomputer is used to control the two DOF positioning device. The PUMA was controlled with an Intel 8086 micorprocessor with an 8087 floating point processor, in conjuction with an Intel ICE 80 development system.

The control software for the positioning device was programmed in MACRO-11 assembly language for fast execution. The sampling times for the experiments were 5ms for the PD MRAC and 7ms for the PID MRAC algorithms. For tests in which the two axes were run simultaneously, the sampling times were twice as long. The PUMA algorithms were programmed in PLM86 and then assembled into 8086 assembly language. The PD MRAC algorithm required 2.3ms for execution, and the PID form required 4.4ms. Both systems used reference models with a bandwidth of 10 rad/sec for PD control. The model bandwidth was selected to eliminate any performance degradations due to sampling time and actuator amplifier saturation. The PID model was third order and was chosen to have 1% overshoot and the same saturation step size as the PD model.

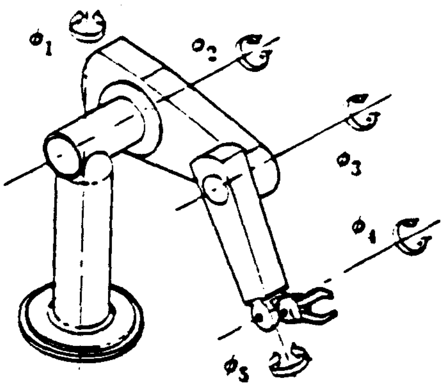

Fig. 3. PUMA Robotic Manipulator.

Simulations of both the table and the robot under adaptive control were run in conjunction with the experimental work. The dynamic model of both the table and the robot were derived assuming rigid links using Lagrange's formulation.

Experimental and Simulation Results

The gains used for the adaptation mechanism, the Q matrix, were bounded by stability analysis. This analysis was also helpful in selecting the values for good performance. Then simulations and the experimental system itself were used to optimize the algorithm's performance.

Fig. 4 shows experimentally measured feedback gains adapting for several sets of adaptive gains. Initially the value of the position feedback gain, for θ_2, (k_{22}) is 5 times lower than that required to match the model. Even for the relatively wide range of Q values, the adaptation can be seen to be quite well behaved. In all three cases k_{22} is close to its final adapted value before the second corner of the profile occurs. The position and velocity errors were reduced by an order of magnitude after the first cycle. The gains in Fig. 4 do not converge to precisely the same value and oscillate about the average value. This effect is due to the nonlinearities in the physical system, such as friction.

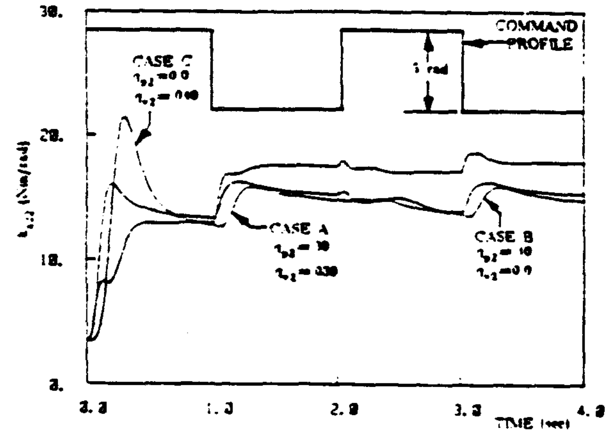

Fig. 4. Effect of adaptive gain selection on the adaptation of joint-2.

As with nearly all adaptive algorithms, the dynamics of the adaptive mechanism are functions of the initial parameter errors and the input. This algorithm is not very sensitive to the intial parameter error as can be seen in Fig. 5. Here, widely differing initial gains are used resulting in very different initial dynamic parameters. The adaptation of the position gain appears uniform in shape and speed.

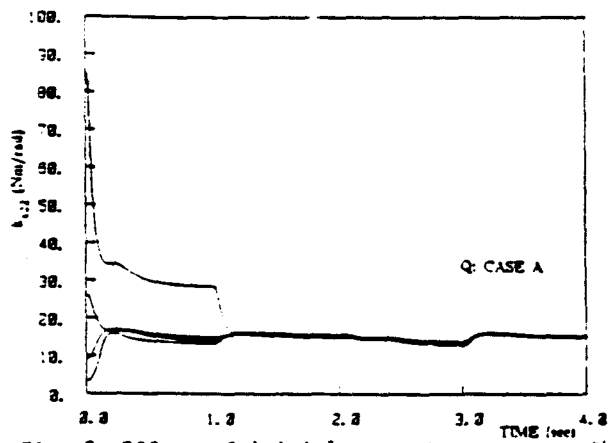

Fig. 5. Effect of initial parameter error on the adaptation of joint-2 for a step input command profile.

The adaptation sensitivity of the system to the input's form and magnitude was also investigated. The magnitude of the input can have a substantial effect on the adaptation performance. As shown in Fig. 6, decreasing the input step size, H, by a factor of two slows adaptation significantly. This effect can be elminated by normalizing Q to the magnitude of the input. Our experimental and simulation results also confirm that the gains can be normalized for a ramp input as well. The sensitivity to the form of the input is more of a problem. This is fundamental to adaptive control. To overcome this problem the use of special learning signals has been suggested [Dubowsky and Desforges,1979]. Since many trajectory planners use low order polynomials and ramps for position reference inputs, it may also be possible to normalize Q for these functions.

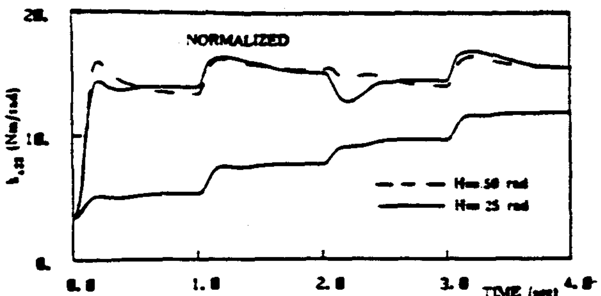

Fig. 6. Effect of input magnitude and Normalization on the adaptation of joint-2 with step command profile.

After selecting the values for Q, the experimental system was used to test the performance of the device under rather extreme conditions. Fig.7 shows an example of the performance of the Ø1 axis when a large payload was added causing the effective inertia of the axis to quadruple. Using feedback gains set for the device with no payload, a large overshoot in the system response results. This response is quite different from the reference model which has been selected to be criticially damped. The system response has been clearly degraded. The adaptive control enables the system to conform to the model by the end of the first step. A similar case for a ramp input is shown in Fig. 8. Here the performance of the system with adaptation is corrected before the first dwell is reached.

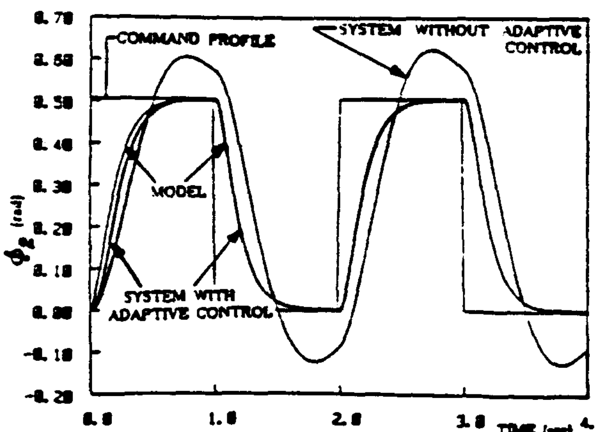

Fig. 7. PD control for joint-2 (with an added payload).

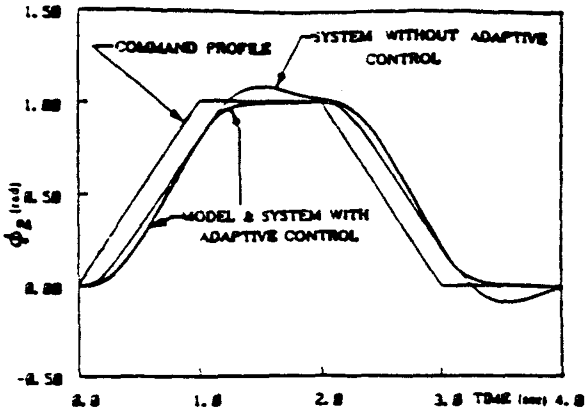

Fig. 8. PD control for joint-2 (ramp command profile) with a payload added.

Fig.'s 8 and 9 are for the motion of a single axis. For many robotic systems (cartesian designs being a notable exception) the dynamics of each axis are highly coupled. This fact is true even for the relatively simple two DOF system used in these tests. To experimentally investigate the adaptive control algorithm's ability to tolerate and compensate for this coupling, a test case was run in which axis 1 was mounted vertically and a payload bar with weights at each end was mounted on to axis 2; with the bar initally vertical (the minimum inertia configuration for joint 1). Joint 2 was commanded to move the payload bar to a horizontal position (the maximum inertia configuration for joint 1) with a ramp profile. At the same time joint 1 was commanded to follow a ramp profile with an amplitude of .5rad at .5rad/s. During this maneuver the effective inertia varies from a minimum to a maximum sinusoidally. Fig. 9 shows the performance for this axis. Without adaptation significant overshoot can be seen while the response with adaption closely follows the model. The maximum position error between the system and the model is 2 1/2 times larger for the nonadaptive case. At the same time the Ø1 axis is adapting, the Ø2 axis is adapting for the added payload. The response for this axis is nearly identical to the single axis result shown in Fig. 8. The error between the system and the model for the Ø2 axis is shown in Fig. 10. Fig. 11 shows simulation results for this case. Comparing these results Fig. 9 clearly shows the ability of the simulation to predict the

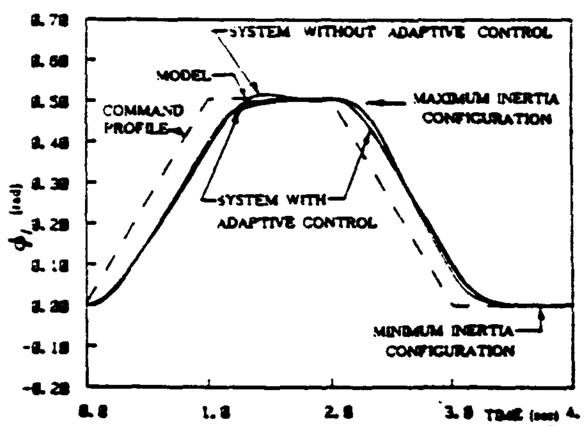

Fig. 9. Experimental performance PD control of joint-2 for multi-axis test case.

experimental results when it is based on accurately measured data.

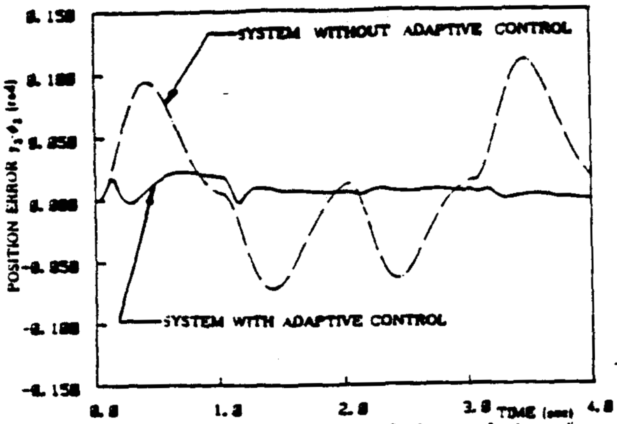

Fig. 10. PD control on joint-2 for multi-axis test case.

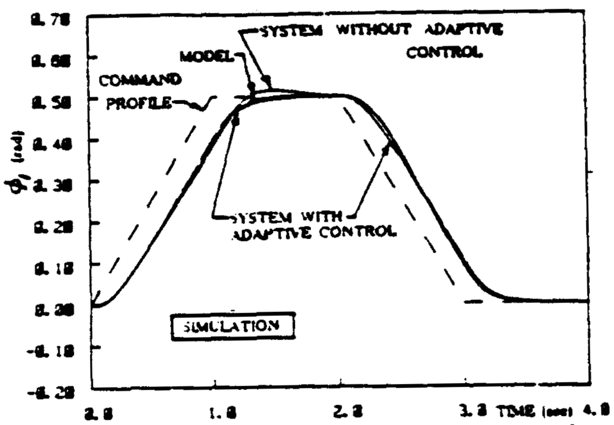

Fig. 11. Simulation performance of PD control on joint-1 for multi-axis test case.

Unmodelled disturbances, such as gravity and joint friction, can degrade manipulator performance. In the above experimental results the joint friction was modelled and used as a dynamic feedforward signal. However, it is not always possible to accurately compensate for joint friction. Integral adaptive control, the PID algorithm, was investigated as an alternative to the feedfoward approach. Fig 12 shows the performance of the PD MRAC without friction compensation as it is used to adapt to a payload change. A large steady state error results. Fig. 13 shows the same case with PID control. This performance closely matches the model, having no steady state error as well as a faster settling time. It would appear that this form of adaptive control offers significant advantages in cases where joint friction is difficult to model.

Fig. 14 shows the result of gravitational disturbances. Here a very large unbalanced payload was attached to axis 2 and axis 1 was tilted so that gravity applied a large varying disturbance to axis 2. Joint 2 was then commanded to move from bottom dead center with a step input. This performance shows serious degradation caused by the disturbance for both the adapting and nonadapting cases. In Fig. 15 the same test is performed with the adaptive PID controller and the error is dramaticaly reduced.

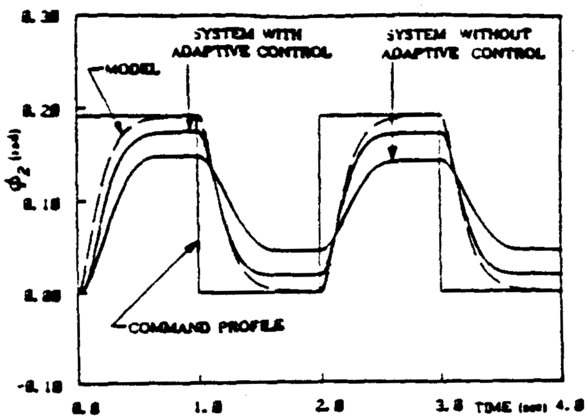

Fig. 12. PD control on joint-2 without Friction compensation.

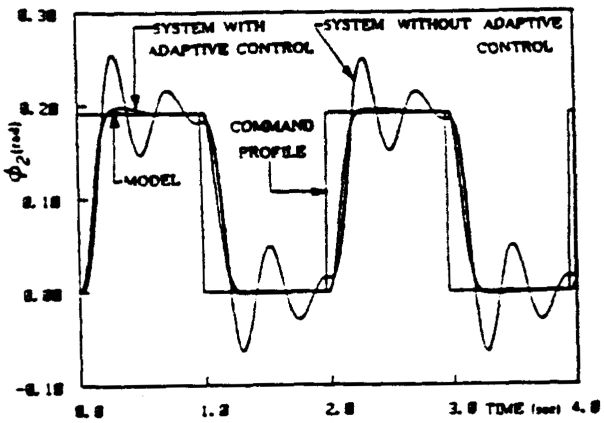

Fig. 13. PDF control on joint-2 without Friction Compensation

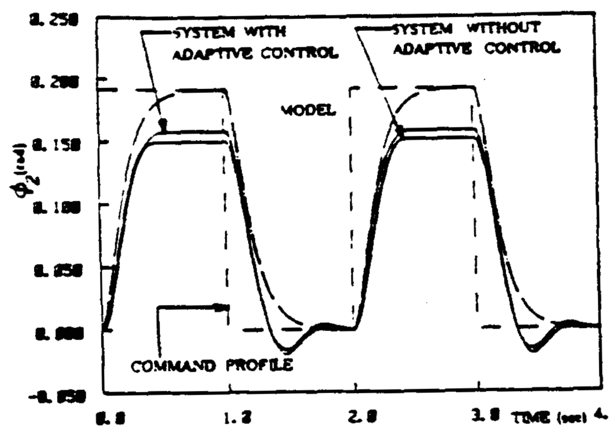

Fig. 14. PD control of joint-2 with unbalanced weight.

The MRAC algorithm was also implemented on the PUMA robotic manipulator. This work was done with the technical cooperation and support of the Westinghouse Corporation, Defense Systems Center, Baltimore, Maryland. The objective was to determine if the properties of commercial quality systems, such as gear train backlash, would degrade the performance of the MRAC algorithm. Many of the tests performed on the two DOF device were repeated using the PUMA with very similar results. The motors of the PUMA operate through very high gear ratios and therefore the effects of friction were relatively small. Hence, no friction compensation was needed.

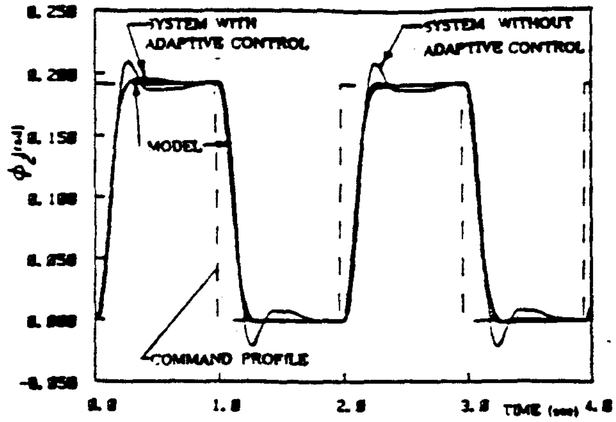

Fig. 15. PDF control of joint-2 with unbalance weight.

Fig. 16a shows the response of the PUMA θ_1 axis when it matches a .707 of critically damped reference model. Fig. 16b shows the axis response to a step command profile without adaptive control, when the feedback gains are set low. For this straight out and horizontal configuration of the manipulator, the response is quite underdamped and does not match the model well. In Fig. 16c the adaptive control is enabled and the response converges to that of the model. In this case the adaptation gains, Q, have been intentionally set quite low so that the changing system response could be observed. The adapting position gain is shown in Fig. 17. The adaptive gains could easily be adjusted so that virtually no differences between the system and the model can be seen , even on the first step. Fig. 18 shows the adapting gains for such a case. Here the initial position gain is lowered by a factor of nearly 6. By the first dwell both gains are nearly at their steady state values. Similar results have been demonstrated for cases where the initial feedback gains were set too high. The gains were reduced smoothly and quickly. Other cases were done in which the manipulator adapted for changes in its configuration and payload. Payloads of up to 10kg were used which drastically degraded the performance of the system without adaptive control. The feedback gains were lowered and raised by the adaptive algorithm, as required to match the system performance to its model.

Conclusions

This paper has shown that an adaptive control algorithm can signficantly improve the performance of robotic manipulators over that obtained using conventional control. The experimental results demonstrate that the algorithm is practical for the control of commercial quality manipulators with relatively modest control computers. It has also been shown that disturbances such as joint friction and gravity, can degraded the performance of manipulator systems and should be considered in the development of their control systems. Results have been shown which demonstrate that these disturbances can be effectively compensated for using an integral adaptive control algorithm. It has also shown that simulations can be an effective aid in the design of manipulator

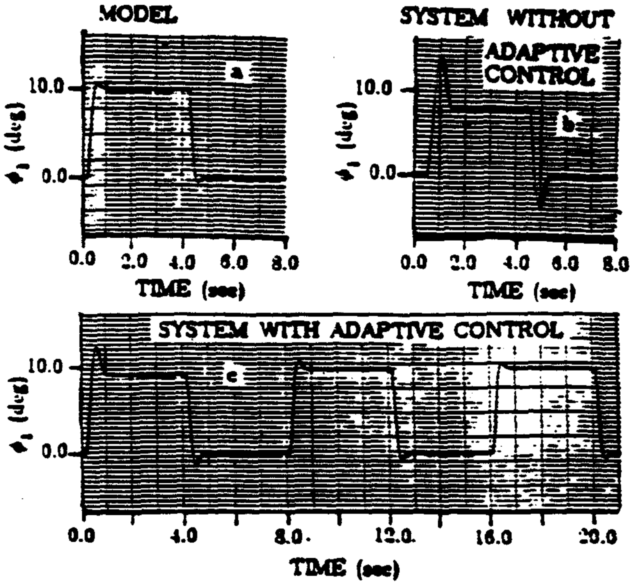

Fig. 16. PUMA response under adaptive and non-adaptive control.

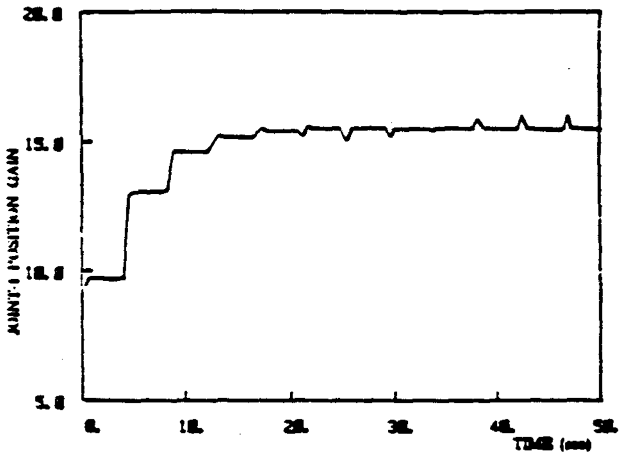

Fig. 17. PUMA slow gain adaptation.

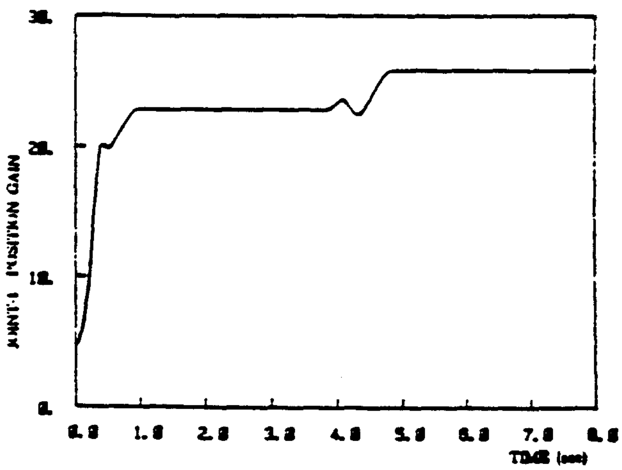

Fig. 18. PUMA fast gain adaptation.

controls when they are used in concert with experimental studies and based on accurately measured system parameters.

Acknowledgements

The authors would like to acknowledge the technical cooperation and support for this study from the Digital Equipment Corp and the Westinghouse Electric Corp. The technical contribution of J. Whalely, of M.I.T., was also invaluable.

References

Bobrow, J.E., Dubowsky, S., and Gibson, J.S., 1983. On the Optimal Control of Robotic Manipulators with Actuator Constraints, Proc. American Cont. Conf., San Francisco, CA

Brogan, W.L., 1974. Modern Control Theory, Quantum Pub., New York, NY.

Donalson, D.D. and Leondes, C.T. 1963. A Model Referenced Parameter Tracking Technique for Adaptive Control Systems, IEEE Trans. on Applications and Industry, 82;68, pp. 241-262.

Dubowsky, S. 1981. On the Adaptive Control of Robotic Manipulators: The Discrete Time Case, Proc.1981 Joint Auto Cont. Conf., Charlottesville, VA.

Dubowsky, S. 1981. On the Dynamics of Computer Controlled Robotic Manipulators, Proc.IV CISM-IFTOMM Sym. On the Theory and Practice of Robots and Manupulators, Warsaw, Poland.

Dubowsky, S., and DesForges, D.T. 1979. The Application of Model Referenced Adaptive Control to Robotic Manipulators, J of Dynamic Sytems, Measurement, and Control, 101:3, pp. 193-200.

Dubowsky, S., and Shiller, Z. 1984. Optimal Dynamic Trajectories for Robotic Manipulators, Proc.V CISM-IFToMM Sym. On the Theory and Practice of Robots and Manipulators, Udine, Italy.

Freund, E. 1982. Fast Nonlinear Control with Arbitrary Pole Placement for Industrial Robots and Manipulators, Intern'l J. of Robotics Research, 1:1, pp.65-78.

Horowitz, R., and Tomizuka, M. 1980. An Adaptive Control Scheme for Mechanical Manipulators - Compensation of Nonlinearity and Decoupling Control, ASME Paper No. 80-WA/DSC-6.

Kahn, M.E., and Roth, B. 1971. The Near-Minimum-Time Control of Open-Loop Articulated Kinematic Chains, J. of Dyn Systems, Measurement, and Cont, 93:3 pp. 164-171.

Koivo, A.J., and Guo, T.H. 1983. Adaptive Linear Controller for Robotic Manipulators, IEEE Trans. on Auto Cont, AC-28:2, pp. 162-170.

Landau, I.D. 1974. A Survey of Model Reference Adaptive Techniques--Theory and Applications, Automatica, 10, pp.353-379.

LeBorgne, M, Ibarra, J.M. Espiau, B. 1981. Adaptive Control of High Velocity Manipulators, Proc. 11th ISIR, Tokyo, Japan.

Lee, C.S.G., and Lee, B.H. 1984. Resolved Motion Adaptive Control for Mechanical Manipulators, Proc. 1984 ACC, San Diego, CA

Luh, J.Y.S., Walker, M. W., and Paul, R.P.C. Resolved Acceleration Control of Mechanical Manipulators, IEEE Trans. on Automatic Control, AC-25:3, pp. 468-474.

National Research Council 1983. Applications of Robotics and Artificial Intelligence to Reduce Risk and Improve Effectiveness: A Study for the United States Army, National Academy Press, Washington, D.C.

Papadopoulos, E.G., 1983. An Investigation of Model Reference Adaptive Control Algorithms for Manufacturing Processes, MS thesis, M.I.T., Dept. of Mech. Engr. , Cambridge, MA

Phelan, R.M., 1977. Automatic Control Systems, Cornell University Press, Ithaca, NY

Raibert, M.H. and Horn, B.K.P., 1978. Manipulator Control Using Configuration Space Method, The Industrial Robot, 5:2, pp. 69-73

Rhors, C.E., Valvani, L., Athans, M., and Stein, G. 1982. Robustness of Adaptive Control Algorithm in the Presence of Unmodelled Dynamics, Proc. 21st IEEE Conf. on Decision and Control, Orlando, Florida.

Takegaki, M., and Arimoto, S. 1981. An Adaptive Trajectory Control of Manipulators, Inter Jour of Control, 34:2, pp. 219-230.

Vukobratovic, M., and Kircanski, N. 1982. An Engineering Concept of Adaptive Control for Manipulation Robots via Parametric Sensitivity Analysis, Bulletin T. LXXXI de l'Academie Serbe des Sciences et des Arts, Classe des Sciences Techniques, 20, pp. 24-39.